Adaptation Strategies:

Using CTODS and a Message Broker to Share Clinical Data on the caGrid

R. Mark Adams, Ph.D.

June 2008

Goal and Agenda

- **Adaptation is a strategy for connecting to the caGrid, enabling collaboration around a range of clinical data. The goal of this session is to briefly outline some adaptation strategies, followed by a particular focus on how institutions can share data by leveraging existing messaging systems.**

- **Agenda:**
  - What is Adaptation
  - Adapting a Tool – Understanding Compatibility
  - Possible Strategies – Design Patterns
  - Decision Process – Some considerations
  - Adapting a Tool – An example in more detail

Setting the Stage

- Connecting with caBIG™ requires tools (e.g., software applications) that are caBIG™-compatible.

- To become caBIG™-compatible, either <u>adopt</u> a tool from caBIG™ or <u>adapt</u> center specific tools to be interoperable with other caBIG™ tools.

- In addition to adopting or adapting tools, these tools will also connect to the caBIG™ infrastructure backbone – called caGrid (or "The Grid"). This is called "establishing a Grid node" (or, "getting on the Grid").

- Once connected to others through the Grid, each tool is able to access caBIG™-compatible tools and data sets at other organizations, as well as offering data sets to share, as appropriate.

Two Pathways – Adopt or Adapt

Adopting a tool involves installing software applications already created by caBIG™, integrating them into the workflow, and connecting to caGrid.

Adapting a tool involves modifying the existing software applications to be caBIG™-compatible, and then connecting to caGrid.

Interface from the Tool to the Grid
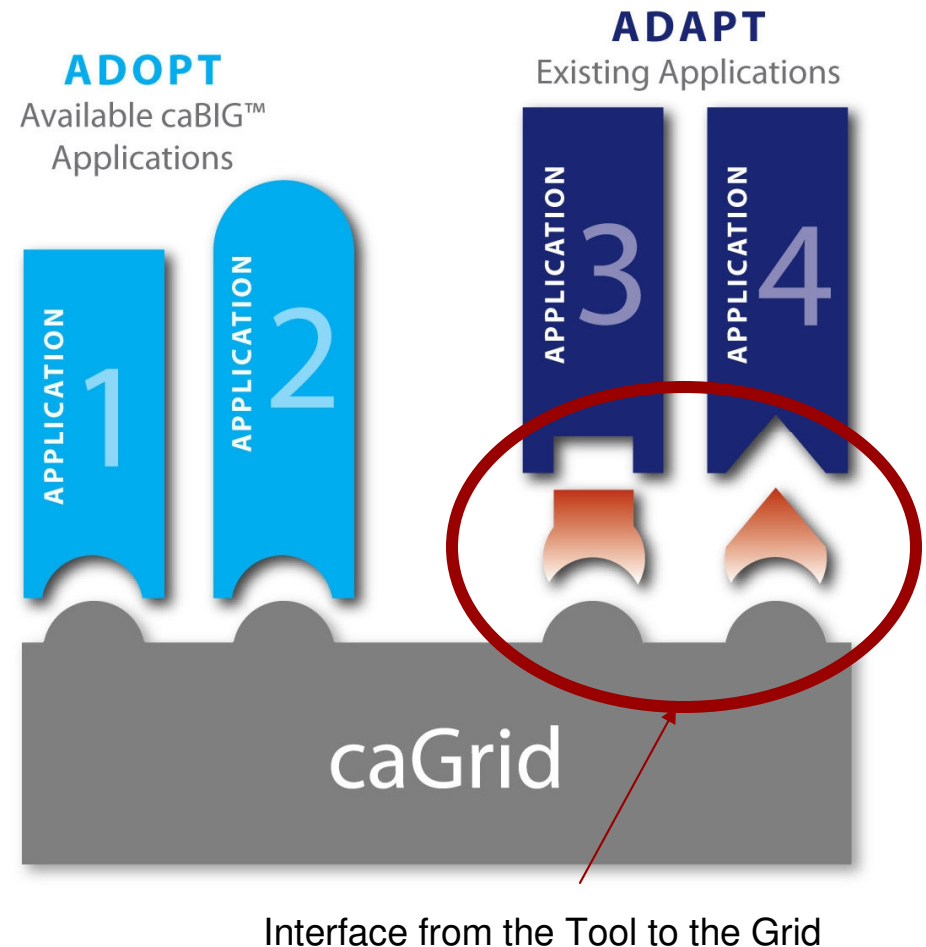
Choosing the Adapt Path

- This presentation focuses on the options associated with **adapt paths** – modifying an existing tool to be caBIG™ compatible.

- Adapting a tool may be the best path if:

  - Tools exist that contain data that is willing to be shared.

  - Existing technology infrastructure and tools meet the needs of users, and are already integrated into workflows.

  - Technical teams have software development expertise and experience with the NCI's vocabulary and data model tools and repositories.

Benefits of adapting:

Access data from other Centers*

Share data with others: both internally and externally*

Access and share data analysis tools and services with others*

Integrate other caBIG™ tools with the compatible tool

Further extend caBIG™ tools to connect with center specific tools

*Also benefits of adopting.

caBIG Cancer Biomedical Informatics Grid

Adapting: There <u>is</u> a Process

1. caBIG™-compatibility begins with an Information (or Data) Model, which represents the interfaces and relationships of a system.

2. The information model is then annotated with Controlled Vocabularies to establish shared meaning across model components (called "semantic integration").

3. This annotated information model is then converted into Common Data Elements (CDEs) that provide the structure (or format) for the data.

4. The information model is also used to generate the Application Programming Interface (API): the mechanism by which data are exchanged.

caBIG™ Cancer Biomedical Informatics Grid™

# Assessing Interoperability: There is a Process For This Too

- Four different levels of caBIG™ compatibility (i.e., interoperability): Legacy, Bronze, Silver, and Gold.

- There are four areas of compatibility - an application must meet the guidelines in <u>all four areas</u> to be considered "caBIG Compatible:"

  - **Syntactic Interoperability**
    - Programming and Messaging Interfaces

  - **Semantic Interoperability**
    - Information Models
    - Common Data Elements (CDEs)
    - Vocabularies and Ontologies
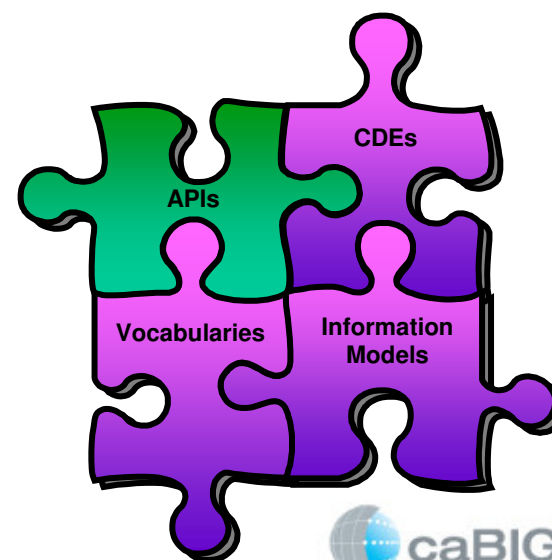
Design Pattern Overview

- The following slides outline **SIX** different approaches for adapting a tool to be caBIG™-compatible. These are conceptual models – foundational strategies - for shaping an adaptation roadmap. This is neither an exhaustive nor mandated list – they are simply examples that we believe can facilitate adaptation based on experience to date.

Understanding Design Patterns

The six approaches offered here are based in the concept of **design patterns**, which Wikipedia defines as "a general reusable solution to a commonly occurring problem in software design."

Design patterns are essentially templates: standardized conceptual solutions that can be applied to many different situations. Based on caBIG™ experiences, there have been six standard ways to approach the software design problems presented by the adapt path identified. These are presented in the following slides.

http://en.wikipedia.org/wiki/Design_pattern_%28computer_science%29

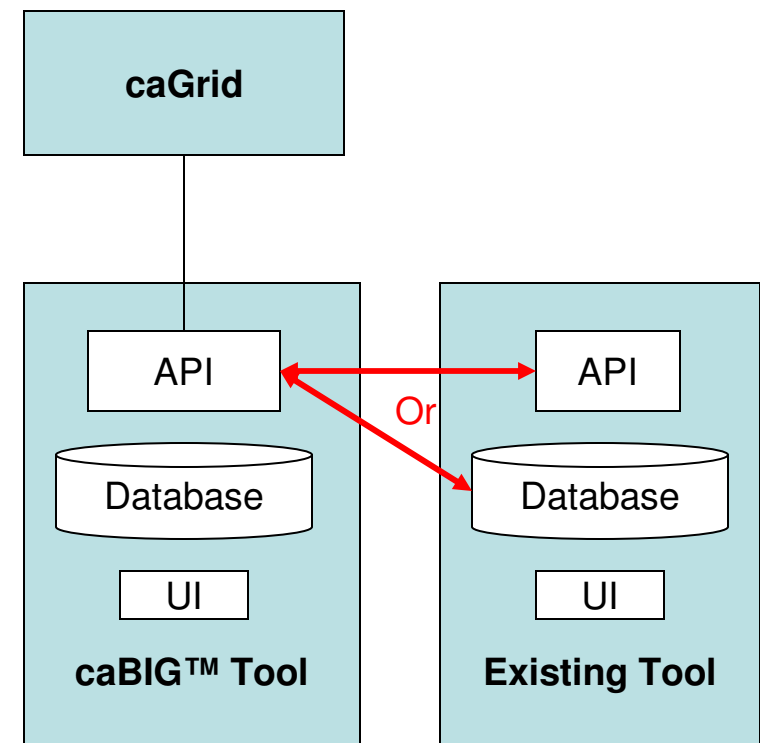caBIG Cancer Biomedical Informatics Grid™

1: Wrapper

- **Use Scenario:** There is a desire to continue using an existing legacy tool, without adopting one from caBIG™. In this case, a new caBIG™ API is generated to allow data exchange between the existing tool and the Grid.

- **Description:** Involves using caCORE to create and semantically map a new caBIG™ compatible API to the existing API. The caBIG™ API then connects to the Grid. This may or may not require a UML model – it could be a field-to-field mapping activity.

caGrid

caBIG™ Compatible API

API

Database

UI

Existing Tool
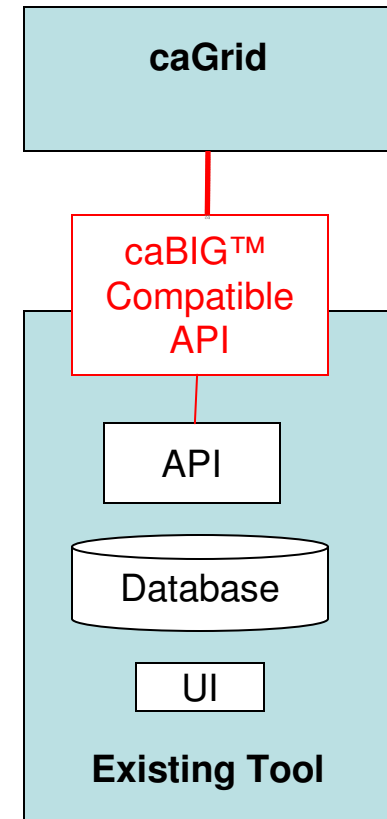
caBIG

2: Direct Data Access (Interim Solution)

- **Use Scenario:** There is a desire or need to retain the legacy database, but also willingness to adopt a caBIG™ tool for its user interface and to facilitate connection to the Grid.

- **Description:** Data "lives" in the legacy database; caBIG™ tool is adapted to query that database (can be dynamic or on demand).

- **NOTE:** This may be an effective transition strategy, but is not recommended as a long-term alternative. May be an interim step leading into the following two patterns.

caGrid

API — Or — API

Database   Database

UI   UI

caBIG™ Tool   **Existing Tool**

caBIG® Cancer Biomedical Informatics Grid®

4: Data Warehouse

- **Use Scenario:** There is a desire or need to retain the legacy database and user interface, but there is a willingness to use a caBIG™ tool to connect to the Grid.

- **Description:** Data "lives" in the legacy database; Extract, Transfer and Load (ETL) techniques are used to periodically import data into caBIG™ tool to serve to the Grid. ETL techniques designed to ensure that data are semantically harmonized.
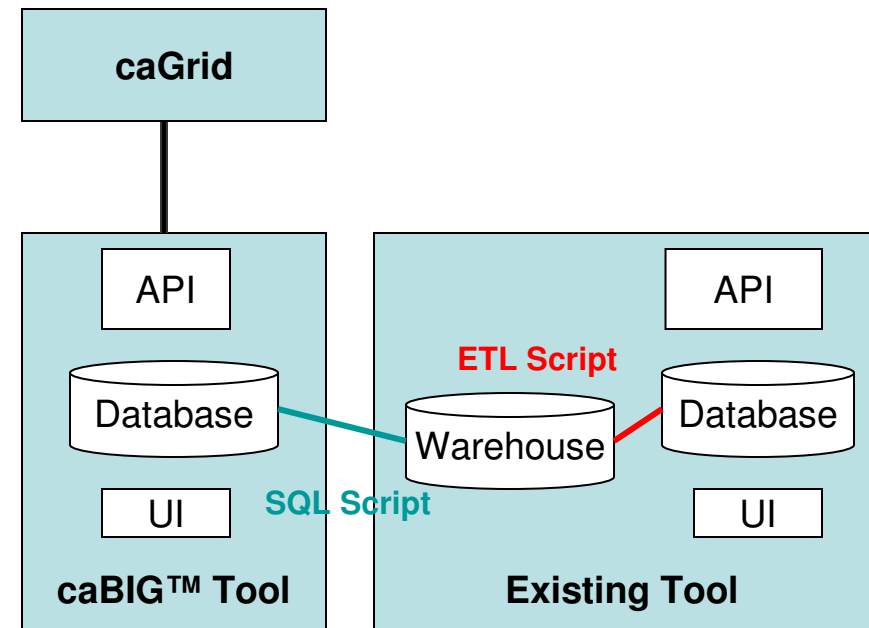
5: Clone and Own

- **Use Scenario:** There is a desire or need to retain the legacy database and user interface, but willingness to use a caBIG™ tool API to connect to the Grid.

- **Description:** An existing caBIG™ compatible API is used and "copied" into the existing tool. This involves mapping legacy schema to the caBIG™ API. Reusing existing caBIG™ CDEs is key to this effort. The center will have to incorporate the changes in the API if there are modifications, changes or upgrades.
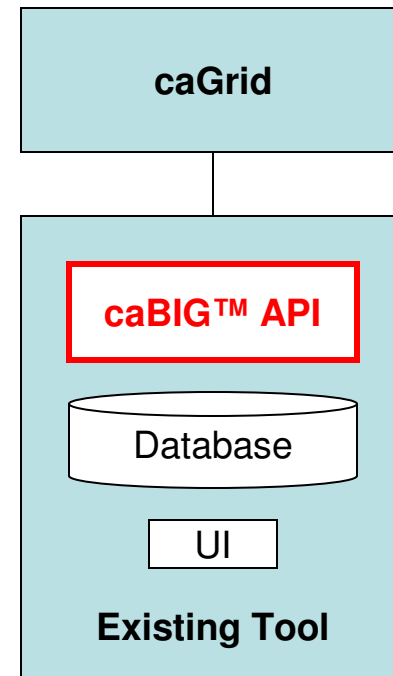

Need to either directly use the caBIG™ application API or develop a duplicate API that operates exactly the same in the application

6: Generate an API

- **Use Scenario:** There is a desire to retain the legacy database, and no existing caBIG™ API is appropriate to map from the legacy database to the Grid. Instead, caBIG™ metadata descriptions and CDEs are used to generate a new API to rest on the legacy tool.

- **Description:** This alternative involves using the caCORE Software Development Kit's (SDK) shared vocabularies and CDEs to construct a brand new caBIG™ compatible API that is mapped to the tables in the existing legacy database.

caGrid

API

Database

UI
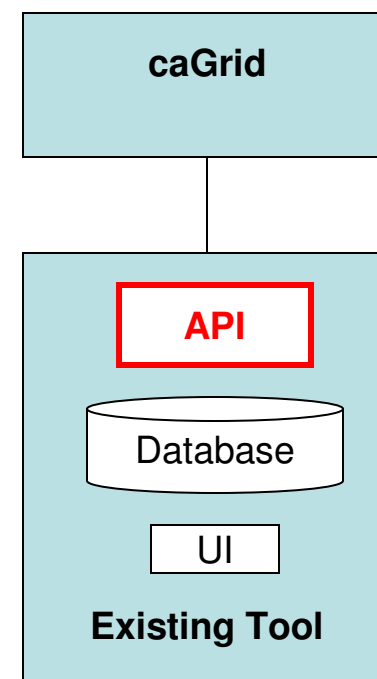
Existing Tool

Use caCORE SDK to generate an API from existing caBIG™ meta-data descriptions and link to legacy database

caBIG

3: Message Broker

- **Use Scenario:** The existing tool already uses standardized messages, such as HL7, to facilitate dynamic information exchange between diverse tools.

- **Description:** A "messaging hub," such as caXchange, brokers the transfer of messages from the existing tools, and feeds into a caBIG™ tool, then out to the Grid. This same concept can be used with other file formats like CSV or tab-delimited ASCII. caAdapter facilitates the mapping into caBIG™ standard elements.

If tools already send messages (e.g., HL7), this may be the most effective way to transmit data from existing tools to caBIG™ grid.

caGrid

| API |
| Database |
| UI |

caBIG™ Tool

API	API
Database	Database
UI	UI

Existing Tools

Message broker hub (caXchange)

caBIG

Variables Impacting Adapt Options

Three variables will help determine the **investment** required to adapt an existing tool – as well as the possible approach that may be most appropriate.

Characteristics of the Existing Tool

Staffing Size, Skills and Budget

Technology Environment

Questions About the Existing Tool

Characteristics of the Existing Tool

- How much visibility is there into the "innards" of the existing tool (e.g., access to data model and underlying structure)?

- How public are the Application Programming Interfaces (API)?

- What data are to be shared? Does it represent a subset of data or the full set?

- The more visibility there is into the tool, the more options there are for adapting it.

- The less visibility or access there is, the fewer options there are for adapting the underlying data structures.

Questions About Staffing & Budget

Staffing Size, Skills and Budget

- Is there access to or funds for an internal or contracted software development team?

- Is there an inter-disciplinary team with domain and technical experts that can be used? Is this the team that actually developed the tool?

- How familiar is the technical team with NCI's caCORE toolset and software development kit?

- How much money can be committed to this process?

- A team with both domain and technical experts will greatly facilitate the process – technical skill sets include data modeling and curation, annotation, familiarity with data model reuse, meta-data reuse, semantic integration, java programming, and database development skills.

- Familiarity with NCI's tools and resources will make some adaptation options easier than others.

caBIG Cancer Biomedical Informatics Grid

Questions About the Environment

Technology Environment

- Who has ownership/control over the network and infrastructure that the adapted tool will eventually live in?

- How many partners are involved in maintaining the environment and regulating data exchange?

- How does this adaptation effort intersect with the overall IT strategy?

- The degree of control over the infrastructure and environment may impact the adapt path that is best.

- The degree to which this adaptation process is connected with other caBIG™ compatibility efforts may also shape the path that is chosen.

Selecting a Design Pattern

	Characteristics of Existing Tool	**Staffing Size, Skills and Budget**	**Technology Environment**
3. Message Broker	• Existing tool has key functionality, and data for the Center. • Don't want or need additional functionality from caBIG™ tool. • Only anticipate sharing a subset of available data.	• Staff has necessary skills, since existing messaging infrastructure exists. • Less expensive than an API, given it is a smaller subset.	• Existing messaging system (e.g., HL7) planned, or in place. • Data consumed from a database as an output, so control over the database is not necessary.

Design Patterns are Translated into Practice

- These Design Patterns represent a high-level outline of a adaptation solutions, and must be fleshed out with details that are specific to their implementation site.

- In order to accommodate the capabilities, limitations and needs of a given site, each implementation will necessarily differ

- As an illustration of what is possible with the caBIG™ tools to facilitate getting clinical data on the caGrid, we will examine the development of an implementation based on Design Pattern #3, "Message Broker"

caGrid

API
Database
UI

caBIG™ Tool

API	API
Database	Database
UI	UI

Existing Tools

Message broker hub (caXchange)

caBIG Cancer Biomedical Informatics Grid
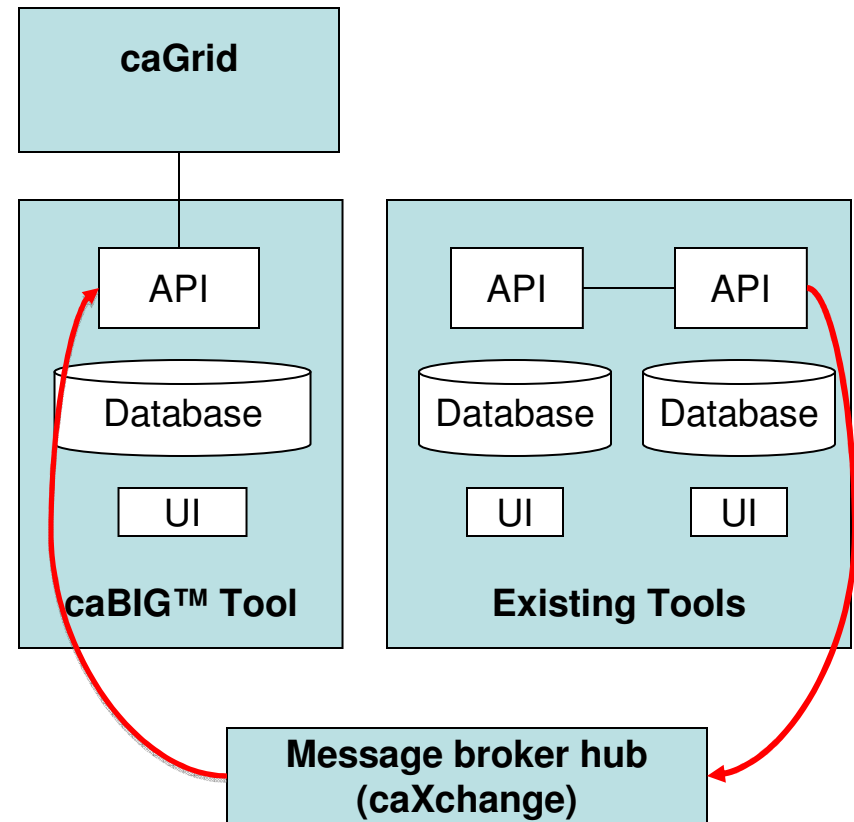
Some Assumptions

- The deploying institution has a need to share data related to a clinical trial with appropriately authorized collaborators, and has chosen the caGrid as the means to do so.

- The data resides within clinical systems that are not readily available for direct access to the database and do not expose usable or accessible APIs.

- The Institution deploying this adaptation strategy has the ability to generate either HL7v2.x messages or *.csv- formatted reports containing the information of interest.

- The CTODS has data elements that correspond to the elements that will be shared

- The deploying institution has either staff or contractors who understand the data elements to be shared well enough to map them to the appropriate data elements in the CTODS data model.

MJH1

caBIG

MJH1   The mapping is really more to the HL7v3 format rather than the CTODS Data Model
             Michael Holck, 4/14/2008

Component Details

- The proposed solution makes use of several components:

How Everything Fits Together

Clinical Data System

An HL7v2.x Or *.csv format Message/file

A Subject Matter Expert maps the data elements between messages

caAdapter

Mapping file to HL7v3
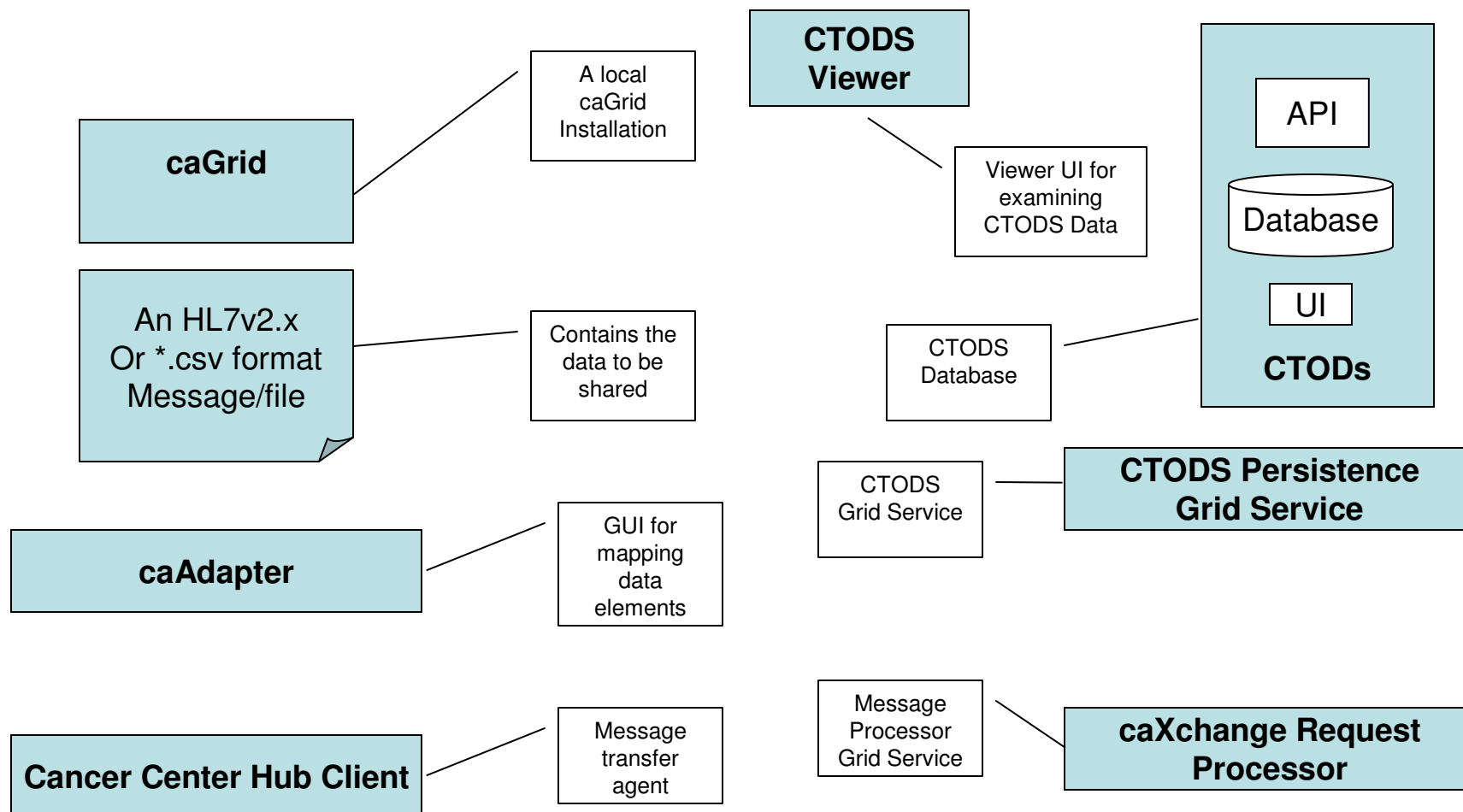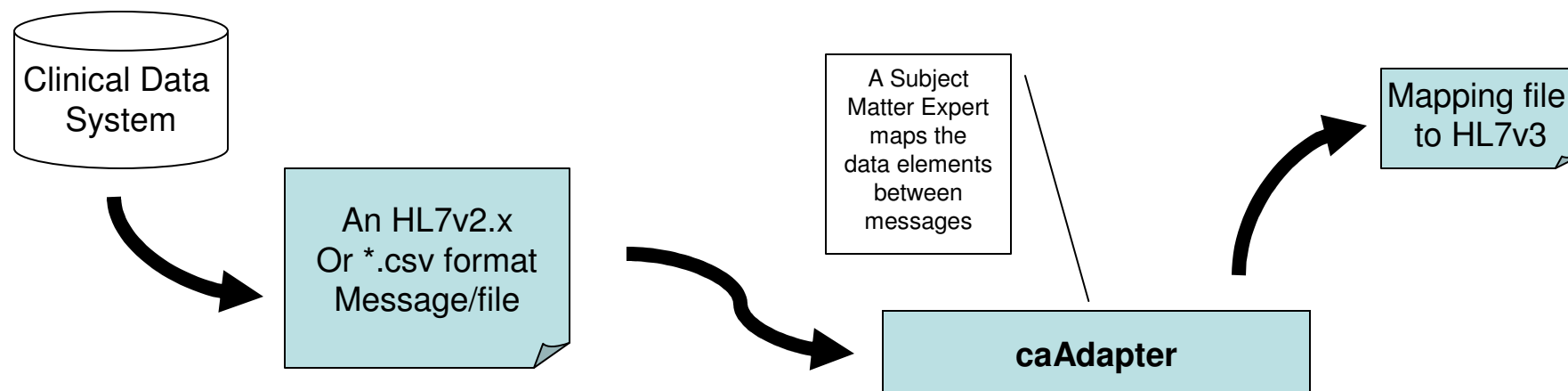
Either a HL7v2.x message is identified or *.csv report generated with the data elements of interest. The data elements from this can be extracted and mapped to those used in CTODS with the help of a GUI tool for that purpose, caAdapter.

caBIG cancer Biomedical Informatics Grid

caAdapter in Action

How Everything Fits Together

Clinical Data System

Generated on-demand, or automatically via triggers or a batch process

An HL7v2.x Or *.csv format Message/file

Mapping File to HL7v3

CCHC transforms the data using caAdapter APIs and submits it to the grid

Cancer Center Hub Client

HL7v3 Message

When the Clinical Data System emits a message, the Cancer Center Hub Client (CCHC) uses the Mapping File to map the data into an HL7v3 message

caBIG Cancer Biomedical Informatics Grid

CCHC in Action

How Everything Fits Together

HL7v3 Message

Automatically manages appropriate messages or files that appear

caXchange Request Processor

caGrid

This service receives the messages from the grid, and adds the data to the CTODS database

CTODS Persistence Grid Service

CTODS Viewer

CTODS

API

Database

UI

The HL7v3 message is then submitted to the caXchange Request Processor, that then routes the message via caGrid to the designated CTODS Persistence Grid Service, which then adds the data to the CTODS database for subsequent query, viewing and analysis

The data can then be queried and viewed by the user

caBIG Cancer Biomedical Informatics Grid

CTODS Lab Viewer in Action

What is Needed?

- The software components described. All are Open Source and freely available.

- The ability to configure and deploy the tools

- Cooperation with and access to the data management resources, so the required data reports/messaged can be used

- Expert mapping of the data elements from the message/report to those used by caBIG/CTODS (HL7v3). This is potentially the most challenging part.

- If the messages being mapped correspond to those already in the tools, this is sufficient, but the tools may need some additional configuration/modification

- An operational and configured caGrid node

caBIG Cancer Biomedical Informatics Grid™

Where to go next

Reference	Link
CTODS	https://cabig.nci.nih.gov/inventory/infrastructure/CTODS
caXchange	**https://cabig.nci.nih.gov/tools/LabIntegrationHub**
HL7 / BRIDG Information	http://www.bridgmodel.org http://www.hl7.org
caCORE Software Development Kit (SDK)	http://ncicb.nci.nih.gov/NCICB/infrastructure/cacoresdk
caDSR Tooling (Semantic Integration Workbench, CDE Browser, UML Model Browser)	http://ncicb.nci.nih.gov/NCICB/infrastructure/cacore_overview/cadsr
caCORE Curriculum	http://ncicbtraining.nci.nih.gov/TP2005/tp2000web.dll/NCICBTraining

caBIG Cancer Biomedical Informatics Grid